ISSN 0112-3351

BEEBLET

CONTENTS

Editorial 2

*FX Calls II: Gurgling Drains 3

ARCHIMEDES: When Greek meets Chinese
    Screendump for Mode 0 or 18 7

Crystal Selection Program 10

Letter to the Editor (1)
    Help Wanted: Wordwise + 18

Peartree RAM Cartridge 20

Surplus Items: Tenders Invited 22

Equipment Division:
    Audited Accounts 24

Windscreen Wipers 26

Classifieds 35 and 28

Letter to the Editor (2)
    Programs, Discs, etc. 29

Control Codes 30

Split Screen Editing 33

Local User Group News
    Wellington 34

Database Services (Advert) 35

Classifieds 28 and 35

Membership 35

Local User Group Meetings, Contacts 36

Beeblet - Contributions and Deadlines 36

Volume 6 Number 3 May 1988

BBC / ACORN

COMPUTER USER GROUP NZ (inc)

EDITORIAL

Beeblet passes a real milestone this month.

This month *Beeblet* contains the first article published in New Zealand specifically for the Archimedes computer user - and it contains the first Archimedes program to be published in New Zealand. We welcome the Archimedes to the fold of Acorn computers.

Almost as exciting is that ice has been broken in that we publish the first article from north of the Bombay Hills since I took over editing *Beeblet* some eighteen months ago. In fact we have two contributions this month. I hope that it is a sign that a thaw has set in and that others will follow suit.

May I thank this month's contributors on your behalf:

John Andrew	Wellington (2)
Paul Collins	Cambridge
Anne Flower	Wellington
David Fung	Wellington
Frank Hazlewood	Auckland
Don McDonald	Wellington
Mike Shores	Auckland
Local User Group	Wellington

Unfortunately the item promised about Melenie has yet to arrive - I hope it will be available for the June issue. Also coming up in that issue is another in Anne Flower's series on *FX Calls and Don McDonald asks: "Anyone for Tennis?"

In the meantime, happy computing.

Graham Ellett

:::

DISCLAIMER

The views of reviewers, contributors and/or Editorial committee are not necessarily shared by

THE BBC/ACORN COMPUTER USER GROUP NZ INC.

:::

*FX CALLS II

GURGLING DRAINS

by Anne Flower, Wellington

GURGLES:

Last time I left you with a few questions about beeps. So far the main uses I can invent for a redefined beep command would be to:

1. Cater for people with hearing difficulties
2. Cut down on typing within a program - instead of SOUND3,-15,124,10 for instance, type V.7 on every occasion, having initialised it at the beginning with the *FX calls.

I would like to hear of any others.

With April Fool belatedly in mind, and assuming you have a BBC Model B, have you come across *FX254,0 - press Break after typing that in. Your computer is now like me - only half there!

On the Master, this command allows the SHIFT key to affect the keys on the numeric keypad:

*FX254,0 and when you press Shift 1 you get ! from the keypad.
*FX254,1 sets things back to 'normal'.

You will probably be aware that a SOUND statement has four parameters (it's In The Book):

Channel, amplitude or envelope, pitch, duration.

We have played around with redefining pitch and duration of the beep, so now let's try to affect channel and amplitude with *FX calls.

*FX211,n will define the channel. Default setting is 3, but perhaps you wanted to use the noise channel:
*FX211,0 will do that.

To force the beep to use a specific amplitude or envelope is a little more involved with *FX212. Because SOUND allows use of H and S, Acorn decided to allow that within *FX212 as well. We're going to ignore that aspect of it.

Amplitude is a negative number in SOUND. For the X register of the *FX212, the value is calculated as:

```
256+((amplitude - 1)*8)
```

If you want it LOUD, the equivalent of -15, that is

```
256+((-15 - 1)*8  =  128
```

*FX212,128 is as loud as you can beep. The default amplitude is the equivalent of -13.

To use a predefined envelope, the value for the X register is calculated as:

```
(envelope# - 1)*8
```

So Envelope 2 would be *FX212,8
and Envelope 5 would be *FX212,48

On the Master, *CONFIGURE LOUD and *CONFIGURE QUIET are alternate ways of getting *FX212,216 and *FX212,240.

That is enough of me sounding off, so on to other things.

DRAINS

Ever had your printer misbehave? Or the computer hang up for some odd reason (SHIFT light on... nothing happening...). Some programs seem to set a printer off.

On other occasions I've had printouts hang up for no apparent reason, and needed some way of releasing the blockage while I try to find out why it's hanging. (Maybe something wasn't plugged in somewhere!!).

While you're trying to figure it out, or get to the rest of the program, is there somewhere to send that stream of printout so that it won't hang up again?

*FX5 has an answer - and *FX5,0 specifically . It sets up a "printer sink" to let all that data drain away safely. If you know in advance that you need this, well and good. However, it would be nice if you were writing a program for others, to detect the problem within it (in case the users haven't read this article), and to accommodate it.

There is OSBYTE 128 which will check any buffer you specify. (ie an *FX that can't be used as *FX...) The idea is that you send a couple of characters to the printer, and check to see if they got there or not. If they didn't, they will still be in the printer buffer, and the program can take action either by way of a prompt "Please turn your printer on" or by setting up a sink for data to go to.

I've said that OSBYTEs can be used from Basic via a PROC or Function or OSCLI, but this one is actually the equivalent of ADVAL which will be simpler to implement.

Thinking ahead, if the printer is in fact enabled, these characters we are going to put into the buffer to test it are going to appear on the paper - I imagine the user won't like that, so we shall send a couple of spaces interspersed with a couple of delete characters to ensure the paper isn't marred:

```
100 DEF 110 PROCtest_printer_availability
120 *FX21,3
130 VDU2,1,32,1,127,1,32,1,127,3
140 IF ADVAL(-4) = 63 THEN PROCprint ELSE PRINT"Please
    enable printer"
150 ENDPROC
```

You may prefer to make line 140 read ... ELSE *FX5,0, or call another PROC to decide which of these two possibilities will be enacted. Line 120 flushes the printer buffer first. There will be more about buffers in a subsequent article.

Used to having a printer plugged in the proper place, but borrowed one that runs off the serial port?

*FX5,2	is what you need before starting to print.
*FX5,1	will get you back to 'normal' when you have your own printer back.

It's also useful if you're rich with a daisywheel hanging off the RS423 port for the professional look, and run a dot matrix off the Centronics port for quick drafts.

As you may have suspected from the terminology above, *FX5,2 is the command to send data down the RS423 port - whether it be to a printer, or another computer; but not to a figment of your imagination - use *FX5,0 for that.

There is more to *FX5, but not today.

With winter coming, dormant printer drivers like *FX123. Are you into writing printer drivers? How about writing this chapter then? And the rest of *FX5.

While we are on printers, if your DIP switches aren't properly set, and line feeds are too many - or perhaps you prefer double-spaced lines (and perhaps double width characters for visibility as well) - and for once you don't want double-spaced printouts.

*FX6,10 is at your command.

*FX6,n tells the computer it mustn't ever let the character specified in register X pass its printer socket (or wherever it actually determines the boundaries). CHR$10 is the Line feed's ASCII value, so *FX6,10 is saying "Ignore all linefeeds. Tell them to get lost."

If you have a Master, it actually responds to *IGNORE n where n is the ASCII value of the character to be ignored.

Friendly isn't it?

Fool around with the listing here during the next month - it will help to have a printer plugged in (or another computer...) to see the effects though.

```
10 VDU2
20 PRINT"The eagles are sedentary"
30 *FX6,101
40 PRINT"The eagles are worse than sedentary"
50 VDU3:END
```

~~~~~~
~~~~~~

WHEN GREEK MEETS CHINESE

or did Archimedes streak backwards.......

BY David Fung, Wellington

Having 1MB of RAM, the Archimedes can apportion up to 320KB of memory for screens. Instead of a fixed starting address for the screen memory, Acorn decreed rather perversely that the starting address should be variable.

In case the users should have too much time to kill, Acorn very considerately omits the information where one may find the screen address so that one may play the game of peek and poke happily for hours. (Subscriptions for "sue Acorn for loss of income fund" is now open)

At the default setting (*CO.Screensize 0) the screen memory starts at &01FD8000 for the Archimedes 310.

In the BBC/Electron/Master the screen memory is arranged from the top to the bottom of an 8 byte column to the top of the next 8 byte column etc in an 80 column row saw-tooth fashion. In the Archimedes the screen memory proceeds horizontally from left to right in rows of 80 bytes (in mode 0) much the same way as one writes.

Working on this principle I wrote a Screendump programme (Listing 1) in BASIC for the H-P Thinkjet printer. It dumps a mode 0 screen in 28.5 seconds. Very impressive, if you like scrambled egg.

On closer inspection it reveals that characters are flipped over 180 degrees so that d becomes b, q becomes p in a dyslexic manner. The artwork is, of course, scrambled. One can applaud Acorn for being consistent in doing things back to front, each screen byte has been written as its mirror image: eg., 11010000 becomes 00001011, and 11101001 becomes 10010111.

Undaunted I added a Procedure to invert (like a sock being turned inside out, not a change of sexual preference) any byte if its value is not 00000000. Listing 2 now dumps faithfully in 65 seconds.

Would some one please find a simpler method or convert mine to ARM assembler codes?

Would Acorn please explain all these mysterious phenomena? It is not Greek, more likely Italian to confound this Chinese I am sure.

Listing 1:

```
 10 REM REVERSEDUMP FOR ARCHIMEDES 310
 20 REM A basic programme to dump mode 0 or 18
 30 REM screen for Hewlett Packett THINKJET Printer
 40 REM BY David Fung, Wellington. March 1988
 50 MODE 0:FORC%=1 TO 2400:PRINTCHR$(RND(223)+33);:NEXT
 51                REM Print a screenful of characters
 60 M%=&1FD8000
 90 TIME=0
100 VDU2
110 VDU1,27,1,42,1,114,1,54,1,52,1,48,1,83;
111                REM set to 640 dots across
120 VDU1,27,1,42,1,114,1,65;
121                REM  starts raster graphic
140 FORI%=1 TO 240
141                REM to print 240 rows of 80 bytes
150   VDU1,27,1,42,1,98,1,56,1,48,1,87;
151                REM send 80 bytes to printer
160   FORB%=0 TO 79
170     VDU1,?M%;
180     M%+=1
190   NEXT
200 NEXT
220 VDU1,27,1,42,1,114,1,66
221                REM finishs raster graphic
240 VDU3
250 PRINTTIME/100:END
```

Listing 2:

```
 10 REM STRAIGHT DUMP FOR ARCHIMEDES 310
 20 REM a basic programme to dump mode 0 or 18 screen
 30 REM for Hewlett Packett Thinkjet printer
 40 REM BY David Fung, Wellington. March 1988
 50 MODE 0:FORC%=0 TO 2400:PRINTCHR$(RND(223)+33);:NEXT
 60 DIM J% 8
```

...... *Archimedes Straight Dump* Contd >>>

...... *Archimedes Straight Dump* Contd >>>

```
 70 FORK%=0 TO 7:J%?K%=2^K%:NEXT
 71           REM put bit values in array of eight cells
 80 M%=&1FD8000
 90 TIME=0
100 VDU2
110 VDU1,27,1,42,1,114,1,54,1,52,1,48,1,83;
111           REM set to 640 dots across
120 VDU1,27,1,42,1,114,1,65;
121           REM  start raster graphic
130 FORA%=1 TO 240
131             REM to print 240 rows of 80 bytes
140   VDU1,27,1,42,1,98,1,56,1,48,1,87;
141             REM send 80 bytes to printer
150   FORB%=0 TO 79
160     IF ?M%=0 I%=0 ELSE  C%=?M%: PROC_I
161               REM if byte value is not 0 invert it
170     VDU1,I%;
180     M%+=1
190   NEXT
200 NEXT
210 VDU1,27,1,42,1,114,1,66
220 VDU3
230 PRINTTIME/100
240 END
250 :
260 DEF PROC_I
261            REM invert the byte to its mirror image
270 I%=0
280 N%=0
290 FORK%=7 TO 0 STEP -1
291              REM Starts from bit 7
300   IF C%>=J%?K% I%+=J%?N%:C%-=J%?K%
301   REM if the byte value C% is larger or equal to the
302   REM (K%)th bit value THEN add the value of the
303   REM mirror image bit (J%?N%) to the value of the
304   REM inverted byte I%. THEN reduce the value of C%
305   REM by the value of the (K%)th bit.
310   N%+=1
320 NEXT
330 ENDPROC
```

CRYSTAL SELECTION PROGRAM

By Mike Shores, Auckland

I have been keenly interest in Electronics for a very long time, and I suppose, like many people tinkering with soldering irons, I've collected one or two quartz crystals along the way. They spring to mind every time I have a need for a frequency standard of some sort.

You don't get very far in Electronics without the ability to make measurements of various sorts and my recent project to make a modem, on the cheap, from basic principles and even more basic i.c's, was no exception: I needed standards of some sort to ascertain that my audio tones for frequency shift keying were correct.

The timebase in my trusty old valve oscilloscope isn't accurate enough and the modem frequencies concerned, by design, bear no easy mathematical relationship to 50Hz mains! Neither are they any kind of product of my standard 1MHz crystal oscillator (successively divided down in decades, eventually to 1sec).

So I started to look again at my pile of crystals; could one of them produce a useful frequency by simply dividing down their output?

In the past I've driven myself crazy with a pocket calculator trying to evaluate all the possible permutations. This always resulted in going out to purchase one or more crystals at great expense in order to get the desired frequency by decade or binary division alone. It takes a mathematical genius to work out the various possible combinations for division of frequency, especially if one tries to incorporate that very practical aspect of allowable tolerance.

So I wrote a program to do it for me.

Astute readers may possibly notice one or two procedures borrowed from other programs, for which I make no apology; I don't believe those kind of short cuts are particularly sinful! Incidentally, neither do I think GOTO's, etc., sinful either. Every program instruction has its place. If you're comfortable with it then by all means use it.

The designers of Basic included instructions to be used.

The following *XTAL* program works it all out by evaluating possible combinations of divisions by 2,3 and 10 as they are relatively easy to arrange and effectively give many other combinations such as 4,6,8,9,12,18 etc.

No doubt other division factors could be incorporated with advantage. This may be the basis of further experimentation on the part of other programmers. Perhaps one possibility is to change the division by ten to division by five.

The program should be carefully typed in its entirety and saved at least twice under different names, different directories, and/or different drive numbers. The "active" program should not be locked as it will re-record itself when first properly run; of course the "safety" back up copies could be locked with advantage.

When you are quite happy all is well, delete line 180 and RUN the program. It should give you a prompt to press function key 0, thus modifying and saving the program so that it can be run again. Thereafter, it will always read the data file *CRYSTAL* where it has stored your various crystal frequencies for future reference.

No really elaborate safeguards are built in as it is assumed that the program will be driven with due care and attention. Such measures could be taken if really necessary but it was considered an unnecessary complication for an otherwise straightforward project.

The names of the procedures are self explanatory.

The program will always list your crystals in order of frequency because of the automatic sort routine initiated when you add to the list. It is assumed that crystal frequencies will be stated in MHz and the required output frequency in Hz. Again, other units could be used by altering the input and calculation routines (PROCinput and PROCcalculation) to suit.

In effect the calculation procedure draws up a complete table of all the permutations of multiple divisions by 3 and 2. It takes each result and looks firstly for a simple match in just the first two significant figures of the target frequency.

If a match is found it goes into PROClikely and compares successive decade divisions of the figure which caused the execution of PROClikely with the target frequency, within the tolerance limits established at the start. Where a match exists the resulting calculation/divisions required, plus the resulting frequency are either displayed on the screen, or printed out if you wish.

You are told exactly which crystal to use and how to divide the frequency to get the required result, thus saving a lot of anguish and probably some money.

- - - ~ ~ ~ - - -

In closing I would strongly urge other readers to contribute programs or articles to the magazine. This article results from a personalised request on the part of the Editor when he kindly replied to a query of mine. My thanks to him for an excellent publication. This article comes from North of Hamilton in "Black Hole" territory. I hope some more Auckland User Group members take the hint.

Ed: Mike's program illustrates another practical way of putting the Beeb to work. Thanks for sharing it with Beeblet *readers.*

You are invited to share developments you have made. Do not assume that no-one will be interested. Or maybe you have a specific problem which you are wondering whether a program has been written to solve; you are welcome to use Beeblet*'s pages to make enquiries.*

....

```
 10 REM ******************************
 20 REM *          'XTAL'            *
 30 REM ******************************
 40 REM *                            *
 50 REM *       By SHORESOFT         *
 60 REM *    (c)  M.W.Shores         *
 70 REM *          1987              *
 80 REM *                            *
 90 REM ******************************
100 ON ERROR GOTO 2270
110 *KEY5*TV0,1¦MMO.3¦M¦S72000OLD¦M¦NL.¦M
120 REM fKey 5 is used to list the program during typing
    in phase.
130 VDU 22,7,23;8202;0;0;0;:REM Mode 7 & suppress cursor.
```

.....XTAL *Contd.* >>>

.....XTAL *Contd.* >>>

```
140 @%=&0090D   :REM Columns 13 chrs. wide.
150 A%=134
160 DIM XTAL(30):REM You can 'DIM' more if you wish!
170 :
180 PRINT '''"Type in full program before deleting"'"this
    line. Then SAVE THE PROGRAM in a"'"different
    directory, or under a"'"different name, BEFORE
    running."'"Just in case ! !":
    PRINT '''CHR$136"KEY 0 can be LETHAL":STOP
190 :
200 N%=2
210 FOR K%=1 TO N%
220 READ XTAL(K%)
230 NEXT
240 :
250 PROCfinish
260 *KEY0 DEL.180,300¦MSAVE"XTAL"¦MRUN¦M
270 PRINT TAB(8)"PRESS KEY 0 TO PROCEED"
280 END
290 DATA 1,2
300 :
310 PROCinit
320 PROCselect_action
330 MODE7       :REM To turn cursor back on
340 PROCfinish
350 END
360 :
370 DEFPROCinit
380 VDU14       :REM Turns on page mode
390 PROCheader("CRYSTAL SELECTION PROGRAM")
400 PROCwindow(0,3,39,21)
410 *FX6,131
420 REM Stops yellow colour code going to printer
430 PROCload_file
440 PROCsort:PROClist
450 ENDPROC
460 :
470 DEFPROCselect_action
480 F%=131:P%=9
490 REPEAT
500 VDU12,15   :REM =CLS & turn page mode off
510 PRINT'CHR$(A%)"0 = Finish"
520 PRINT CHR$(A%)TAB(20)"1 = List crystals"
530 PRINT CHR$(A%)"2 = Add crystal"
540 PRINT CHR$(A%)TAB(20)"3 = Delete crystal"
```

.....XTAL *Contd.* >>>

.....XTAL *Contd.* >>>

```
550 PRINT CHR$(A%)"4 = Enter req. freq."
560 REPEAT
570 PRINTTAB(26,P%)"  "
580 PRINTTAB(6,P%)CHR$(F%)
590 INPUTTAB(7,P%)"Enter your choice ";REPLY$
600 REPLY$=LEFT$(REPLY$,1)
610 UNTIL REPLY$>="0" AND REPLY$<="4"
620 CLS:IF REPLY$="1" PROClist
630 IF REPLY$="2" PROCadd_crystal
640 IF REPLY$="3" PROCdelete_crystal
650 IF REPLY$="4" PROCenter_freq
660 UNTIL REPLY$="0"
670 ENDPROC
680 :
690 DEFPROCload_file
700 F1=OPENIN("CRYSTAL")
710 INPUT #F1,N%
720 FOR K%=1 TO N%
730 INPUT #F1,XTAL(K%)
740 NEXT
750 CLOSE #F1
760 PRINT TAB(10)"CRYSTAL FILE READ"''
770 ENDPROC
780 :
790 DEFPROCsort
800 LOCAL i,posnsmallest
810 FOR i=1 TO N%-1
820 PROCfindsmallestentryfrom(i)
830 PROCswap(i,posnsmallest)
840 NEXT
850 ENDPROC
860 :
870 DEFPROCfindsmallestentryfrom(i)
880 LOCAL next
890 posnsmallest=i
900 FOR next=i+1 TO N%
910 IF XTAL(next)<XTAL(posnsmallest) THEN
    posnsmallest=next
920 NEXT next
930 ENDPROC
940 :
950 DEFPROCswap(k,l)
960 LOCAL temp
970 temp=XTAL(k):XTAL(k)=XTAL(l):XTAL(l)=temp
980 ENDPROC
```

.....XTAL *Contd.* >>>

.....XTAL *Contd.* >>>

```
 990 :
1000 DEFPROCfinish
1010 PROCheader("All done")
1020 F1=OPENOUT("CRYSTAL")
1030 PRINT #F1,N%
1040 FOR K%=1 TO N%
1050 PRINT #F1,XTAL(K%)
1060 NEXT
1070 CLOSE #F1
1080 PRINTTAB(7,12)CHR$(A%)"New CRYSTAL file saved"
1090 @%=10
1100 *KEY 0
1110 REM clears fKey0
1120 *KEY 5
1130 REM clears fKey5
1140 ENDPROC
1150 :
1160 DEFPROClist
1170 PRINT'
1180 FOR K%=1 TO N%
1190 PRINT K%,XTAL(K%)
1200 NEXT
1210 PROCwait
1220 ENDPROC
1230 :
1240 DEFPROCadd_crystal
1250 PRINT'''"Frequency of crystal"
1260 INPUT"to be added to list ";X
1270 N%=N%+1
1280 XTAL(N%)=X
1290 PRINT'TAB(9)CHR$132"The list now reads:- "
1300 PROCsort:PROClist
1310 ENDPROC
1320 :
1330 DEFPROCdelete_crystal
1340 PROClist
1350 INPUT'''"Number of crystal to be deleted ";C%
1360 IF C%=N% GOTO1410
1370 REPEAT
1380 XTAL(C%)=XTAL(C%+1)
1390 C%=C%+1
1400 UNTIL C%=N%
1410 N%=C%-1
1420 CLS:PROClist
1430 ENDPROC
1440 :
```

.....XTAL *Contd.* >>>

.....XTAL *Contd.* >>>

```
1450 DEFPROCenter_freq
1460 C$=CHR$(F%)
1470 LOCAL X:answer=FALSE
1480 PRINT''C$;"Do you require a printout ?":
     IF FNy_n VDU2 ELSE VDU14
1490 PRINT'''C$;"What frequency are you"
1500 PRINT C$;"interested in (Hz)"
1510 REPEAT
1520 PRINT''C$;:INPUT SPC(20);W
1530 UNTIL W>0
1540 PRINT'C$;"What tolerance"
1550 PRINT C$;"is acceptable"
1560 PRINT C$;"( + or - percent )"
1570 PRINT''C$;:INPUT SPC(20);tolerance
1580 lowest=W*((100-tolerance)/100):
     highest=W*((100+tolerance)/100)
1590 *FX3,4
1600 CLS
1610 PRINT TAB(6)CHR$130"Processing.....Please wait"
1620 *FX3,0
1630 PROCcalculation:PROCwait
1640 ENDPROC
1650 :
1660 DEFPROCcalculation
1670 W$=LEFT$(STR$(W),2)
1680 FOR K%=1 TO N%
1690 div_three%=-1
1700 f_loop0=XTAL(K%)*10^6
1710 REPEAT
1720 div_three%=div_three%+1
1730 f_loop1=f_loop0/3^div_three%
1740 PROCtwos
1750 UNTIL f_loop1<W
1760 NEXT K%
1770 VDU 3,15 :REM Turn off printer, turn off page mode.
1780 IF answer=FALSE PRINT'TAB(12)"NONE SUITABLE"
1790 ENDPROC
1800 :
1810 DEFPROCtwos
1820 div_two%=-1
1830 REPEAT
1840 div_two%=div_two%+1
1850 f_loop2=f_loop1/2^div_two%
1860 IF LEFT$(STR$(f_loop2),2)=W$ THEN PROClikely:
     REM two most significant digits do match
```

.....XTAL *Contd.* >>>

.....XTAL *Contd.* >>>

```
1870 UNTIL f_loop2<W
1880 ENDPROC
1890 :
1900 DEFPROClikely
1910 div_ten%=-1
1920 LOCAL f_test
1930 REPEAT
1940 div_ten%=div_ten%+1
1950 f_test=f_loop2/10^div_ten%
1960 IF f_test<highest AND f_test>lowest PROCrecord
1970 UNTIL f_test<lowest
1980 ENDPROC
1990 :
2000 DEFPROCwait
2010 PRINT''TAB(7)CHR$129;CHR$136"P R E S S    S P A C E"
2020 REPEAT UNTIL GET=32
2030 ENDPROC
2040 :
2050 DEFPROCrecord
2060 answer=TRUE
2070 PRINT'''"Crystal No.";K%;", ";XTAL(K%);" MHz."
2080 L$=STR$(XTAL(K%)*10^3)+" KHz"
2090 IF div_ten%>0 M$="/10^"+STR$(div_ten%):L$=L$+M$
2100 IF div_two%>0 M$="/2^"+STR$(div_two%):L$=L$+M$
2110 IF div_three%>0 M$="/3^"+STR$(div_three%):L$=L$+M$
2120 L$=L$+"="+STR$((((XTAL(K%)*10^6)/10^div_ten%)/
     2^div_two%)/3^div_three%)+" Hz"
2130 PRINT L$
2140 ENDPROC
2150 :
2160 DEFPROCwindow(X%,Y%,W%,H%):REM start,width,height
2170 VDU28,X%,Y%+H%-1,X%+W%-1,Y%
2180 ENDPROC
2190 :
2200 DEFPROCheader(A$):VDU12:FORI%=0 TO 1:
     PRINTCHR$&81;CHR$&9D;CHR$&83;CHR$&8D;:
     PROCcenter(A$,VPOS):NEXT:ENDPROC
2210 :
2220 DEFPROCcenter(A$,vpos)PRINTTAB((20-LENA$DIV2),vpos);
     A$:ENDPROC
2230 :
2240 DEFFNy_n:LOCALI%:REPEAT:I%=INSTR("YyNn",GET$):UNTILI%
2250 IFI%<3=TRUE ELSE=FALSE
2260 :
2270 CLOSE#F1:@%=10:IF ERR=17 RUN ELSE REPORT:
     PRINT" at line ";ERL:END
```

~~~~~~
~~~~~~

LETTER TO THE EDITOR

HELP WANTED — WORDWISE +

Dear Editor,

I know feedback - or lack of it - is the bane of *Beeblet's* contributors. For instance, where are the *Electron Column*, the *Beginners' Column*; will we hear from Tony Wattam or Sadie Ash ever again? And... the *HELP Column*?

I need that right now:

It seems as though the WORDWISE + language should be accommodating enough for one to include screen dumps within one's text. Not in the text file necessarily, but on the printout. I can't do it. Could someone please rise to the challenge and provide the program(s) to do it?

This is what I think needs to happen:

1. Set up on disc the screen dump - presumably by having the picture on screen, and *SPOOL pic_dmp to open a disc file just before calling the printer dump routine then *SPOOL to close the disc file at the end.
2. In a segment set up all the variables for the start of the text - line spacing (as a printer code!), PL, LL, TS, BS, DH, DF, RPS's, EP etc., pica or elite and other OCs.
3. At the appropriate point(s) in the text do a CP. Say a screen dump takes up the equivalent of 24 text lines. In fact the printer may do, say, 32 lines to produce the picture, but they will be spaced much closer together.
4. Either increase PL by the difference between 24 and 32 lines for this page, or at least store the number of lines already used on the page - I don't want a dump being split across two pages because WORDWISE thinks 32 lines take more than they really do.
5. Get File of the picture dump and action it.

6. Reset PL or lines used to what they would have been if only text had been printed.

7. Reset all sorts of other things - my printer doesn't use Epson-alike codes, so there isn't the equivalent of ESC @ to reset the printer at the end of a dump - so line spacing in ESC codes, pica or elite (as specified in the start-up segment) - or else simply re-use the set-up segment with an incremented page number so it doesn't (eg) call page 5 of text page 1 again.

While I'm asking for WORDWISE + help, what is the program to put edge-justified page numbers into a DH or DF - that is, right-justify odd-numbered pages and left-justify even-numbered pages. What if I want them extra large using Shift Out (OC14). The left-justified ones are ok, but for the right-justified ones presumably you have to halve the LL, print the page number then reset LL to normal??

I look forward to someone, somewhere, helping on this..

Anne Flower

~~~~~~

Editor: If you can assist Anne please address your reply to me, PO Box 9592, Wellington. The answer is likely to interest other WW+ users.

Although the HELP Column *has been discontinued, this was mainly through lack of questions. Send in your questions - without a formal* Help Column, *I will publish them as above and request assistance.*

If the quantity of questions justifies the HELP Column *being reinstated then that will be arranged. Please be patient for a reply - I cannot reduce the publishing cycle.*

~~~~~~

* PEARTREE RAM CARTRIDGE *

Extra 32K sideways RAM for the BBC Master

by Paul Collins, Cambridge.

Along with many others I have over time acquired several ROMs for my trusty Beeb of varying usefulness. After the first ROM the familiar problem arose of which one to leave in the machine; the acquisition of a Master 128 seemed to be the answer.

However the price of cartridges soon led to the use of ROM images loaded from disc. Eventually even the four standard slots proved insufficient. The answer came in the form of Peartree's MR7200 32K RAM cartridge when a local agent started clearing out before relinquishing the Acorn agency (shame).

Packaged in a standard sized ROM cartridge it has a switch on top to switch write protection on/off and another hole beside it marked ROM A or B. Presumably, Peartree used the same housing for another ROM cartridge.

Fitting it was just a case of pushing it into place and booting the accompanying disc. The menu displayed the equivalent of a *ROMS showing two of the original ROM slots were now configured as RAM. Other options permitted clearing. testing. loading and saving the RAM.

From this menu it is also possible to *CAT a disc, issue a * command and edit the sideways RAM in Hex or ASCII. Fortunately, these commands are provided as a ROM image allowing their use without saving your current work and booting the disc. To use it requires switching to the ROM filing system and, as yet, I have not found an easy way of *CATing *LOADing and *SAVEing to a disc using this method.

The only other software provided is a 'Modern' character font which, while interesting, will probably be consigned to the if I ever need it box as I didn't find it particularly pleasing to the eye.

Playing around with the extra RAM led to the inevitable crash requiring reloading various files I was using before I went poking! where I shouldn't ought to. This time consuming chore resulted in the following files on a disc with images of all my favourite ROMs reducing the keystrokes to a minimum.

```
 *BUILD !BOOT <ret>
   1*BASIC
   2*SRLOAD <filename> 8000 n Q (where n is the slot
                                        number)
   3CHAIN"MESSAGE"
   4 <escape>

*OPT 4,3 (to set shift+break to *EXEC !BOOT)


   10 MODE 7
   20 PRINT CHR$(137)CHR$(131)"PRESS CRTL + BREAK"
   30 PRINT'' To initialise sideways ROMS "
   40 END

SAVE"MESSAGE"
```

Although simplifying the reboot process, a better idea would be for the !BOOT file to chain a ROM loader program to avoid rebuilding the !BOOT file each time you wanted to load another ROM.

Over the last two weeks the cartridge has proved to be ideal for the activities here allowing up to six sideways ROMs to be played with, saving wear on the Master's cartridge slots, not to mention eliminating the risk of bending pins over while inserting ROMs into their sockets.

For occasional ROM users the Peartree RAM Cartridge would not appear to be particularly good value at a NZ RRP of $129. However for the dabbler in sideways who obtains supplies direct from UK it becomes worth considering.

On the matter of size, it is interesting to note the release in UK recently of 256K RAM cartridges configurable as disc drives. Now there's a thought. 256 mode 7 screens on silicon disc would give 10 seconds of cartoons!!!

Who's game to try ?

::

TENDERS TENDERS TENDERS

ARE INVITED FOR THE
SURPLUS ITEMS
LISTED BELOW:

Please send your tender (and Phone No.) to
D. Fung, P.O. Box 28-023 Wellington 5
Tender closes 2 weeks after mailing of Beeblet
The highest tender may not necessarily be accepted.

NEW ITEMS WITH WARRANTY:

1 BBC Master 128 Microcomputer
1 Turbo_coprocessor
1 Thomson 12" Green Monitor (VM 3102 VG)
16 boxes of 10 3M 5.25" DSDD discs
  (any number of boxes up to 16)

……

USED ITEMS:

1 BBC Model B Microcomputer with DFS
1 Opus Challenge 3 Disc Drive
  (1 Double sided floppy drive
    + 1/2 MB Ramdiscs)
1 ADE-55 14" RGB Monitor
1 6502 Second Processor
1 Solidisk 256K Ramdisc board issue 2

……

EVALUATION ITEMS BUT RARELY USED:

1 Acornsoft Iso Pascal + Reference books
1 Minerva System Delta + Index Card
  + Reference Manuals

……

BOOKS: MOST ARE NEARLY NEW CONDITION:

For the BEEB:

The BBC computer book - 1982
Basic programming, Cryer & Cryer - 1982
Programming tips, Acorn User - 1983
Programming the BBC Micro, Williams - 1983
The BBC Micro Book, McGregor and Watt - 1983
Advanced Programming Techniques, McGregor & Watt - 1983
The Advanced User Guide, Bray, Dickens & Holmes - 1983
Basic Rom User Guide, Plumbley - 1984
Assembly Language Programming, Birnbaum - 1982
Beyond Basic, Freeman - 1983
Advanced Machine Code Techniques, Stephenson & Stephenson - 1984
Applied Assembly Language, Edward Ball - 1984

For the Master 128

Reference Manuals - Parts one & two - 1986

General

Yours first Computer, Zaks - 1980
Programming the 6502, Zaks - 1980
Starting Forth, Brodie - 1981
Introduction to word processing, Glatzer - 1981
Pascal for Students, Kemp - 1982
Pascal-User Manual and report, Jensen & Wirth - 1974
System Analysis & Design, Leeson - 1985

****

Please send your tender (and Phone No.) to
D. Fung, P.O. Box 28-023 Wellington 5
Tender closes 2 weeks after mailing of *Beeblet*
The highest tender may not necessarily be accepted.

****

BBC/ACORN COMPUTER USER GROUP NZ (Inc)

EQUIPMENT DIVISION

The Statement of Accounts and Balance Sheet as published last month were unaudited and were subject to actions taken at the AGM. For details, see the Minutes of that meeting and the Note below.

STATEMENT OF ACCOUNTS 1987

INCOME

1986 $		SALES $	STOCK 31.12.87 $	TOTAL $
7308	Supplies	1883	1004	2887
11048	Equipment	605	295	900
7	Sundries	23		23
90	Interest	447		447
18453	TOTALS	2858	1299	4257

EXPENDITURE

1986 $		STOCK 1.1.87 $	PURCHASES $	COSTS $
7539	Supplies	2019	616	2635
10566	Equipment	1288		1288
69	Sundries		58	58
18174	TOTALS	3307	674	3981
189	Profit on Sales		-171	
279	Surplus for Year		276	

NOTE:

Stock to the value of $380 has been written off as a stock loss as approved by the Annual General Meeting. This is reflected in the overall loss in sales transactions for the financial year. The loss was offset by the income derived from investments giving a resultant surplus for the year of $276.

BBC/ACORN COMPUTER USER GROUP NZ (Inc)

EQUIPMENT DIVISION

BALANCE SHEET 1987

0	Member's Funds 1.1.87	279		
279	Plus Surplus		276	
279	ACCUM. FUNDS		555	555

Represented by :

1986	LIABILITIES	1987	1986	ASSETS	1987
$		$	$		$
4000	Bal Loan	2000	3183	Cash Bank	3221
2154	Adv Stock	2154	0	Cash Hand	191
			3307	Stock	1299
58	Creditors	43	0	Debtors	41
6211		4197	6490		4752
	Less Liabilities..		(6211)		4197
NET WORKING CAPITAL			(279)		555

AUDITOR'S STATEMENT:

I have examined the financial records of the BBC/Acorn Computer User Group Equipment Division for the year ended 31 December 1987. I have obtained sufficient information and explanations concerning the account. According to those records and explanations, the above Statement of Accounts and Balance Sheet show a true and fair view of the Equipment Division's financial operations for the year ended 31 December 1987 and its financial position at that date.

Signed: A. Flower (Hon. Auditor)

WINDSCREEN WIPERS

by D S McDonald, Wellington

BeeBug magazine, Vol 6(7), Dec 1987, contains a program by Joan Barnard to make snow-flakes / snow storm. I have written a program to simulate rain drops running down a window. It includes gravity, wind, friction, shape, size, coalescence, and random effects.

The program given here, WIPERS, is shorter than my rainy window program, which was for another computer. Program WIPERS simulates rain or snow falling on a car windscreen and being wiped away by the windscreen wiper. All the while rain continues falling.

The data required are:-

1. rate of precipitation,
2. wiper speed,
3. period of wiper, and
4. half-angle of wiper arc.

A tip - when running a program with the same small amount of keyboard data each time, store the data items on a function key with carriage returns etc. If possible do not use data values which end in 0 or line numbers, because if the program halts an unused data item such as "10" "20" will delete lines 10, 20 and you may not notice it missing. BBC User Guide, p. 142.

I have left the program short so that there is less for you to type in and you can make your own changes and enhancements without having to figure out miles of statements.

The density of rain spots on the surround of the wiper area is uniform. It accumulates for two minutes and then the display is cleared. You can see the wiper moving and raindrops immediately start falling on the dry area.

The density of drops on the wiper area is proportional to the time since the wiper passed that point. My eye was not a very good judge of the density; if the density was two to four times greater in the surround, I could not easily judge that there was a wiper area in the centre that had received much fewer drops.

The best time to press ESCAPE is two-thirds through a wiper sweep.

```
 10 MODE7
 20 PRINT"PROG WIPERS BY D.MCDONALD, THU 10.3.1988"'
    "SIMULATES RAINDROPS ON CAR WINDSCREEN AND WINDSCREEN
    WIPERS"'"CAN ADD WIND, GRAVITY, SHAPE, COLOUR,
    COALESCENCE, RIVULET AND RANDOM EFFECTS."
 30 PRINT'"PRESS ANY KEY TO CONTINUE OR f7 FOR
    EXAMPLE"
 40 *KEY7"X19|M9|M5|M24|M"
 50 A$=GET$
 60 INPUT"ENTER PRECIPITATION RATE 1-20",PRECIP
 70 INPUT"ENTER WIPER SPEED (10 DEG)",SPEED
 80 INPUT"ENTER WIPER PERIOD (5 SEC.)",PERIOD
 90 INPUT"ENTER WIPER ANGLE (25 DEG.)",ANGLE
100 X=-1 : REM INITIALLY WIPE FROM LEFT TO RIGHT
110 PRINT''"WRITE DOWN DATA. VARIABLES PRECIP, SPEED,
    PERIOD, ANGLE."''"PRESS ANY KEY TO CLEAR SCREEN AND
    CONTINUE."
120 A$=GET$
130 MODE5:REM 4 COLOURS, 160*256 PIXELS
140 GCOL0,3:GCOL0,128
150 PRINT"PRECIP";PRECIP"SPEED"SPEED"PERIOD"PERIOD
    "ANGLE"ANGLE
160 MINUTES=2
170 CYCLE=TIME+MINUTES*6000
180 REPEAT
190 SWEEP=TIME+100*PERIOD
200 REPEAT
210 PROCDROPS(PRECIP)
220 UNTILTIME>SWEEP
230 X=-X : REM REVERSE WIPER DIRECTION
240 PROCERASE(900,150,-600,-X*ANGLE,X*ANGLE,X*SPEED)
250 UNTILTIME>CYCLE
260 GOTO130
270 END
280 :
290 DEFPROCDROPS(N)
300 FOR DROP=1TON
310 PLOT64+1+4,RND(1279),RND(1023)
320 NEXTDROP
330 ENDPROC
340 :
350 DEFPROCERASE(Top,Bottom,Centre,A1,A2,SPD)
360 FORA=A1+SPD TOA2 STEPSPD
```

.....*WIPERS* Contd >>>

.....*WIPERS* Contd >>>

```
370 MOVEFNX(Top,A-SPD),FNY(Top,A-SPD)
380 MOVEFNX(Bottom,A-SPD),FNY(Bottom,A-SPD)
390 PLOT80+4+3,FNX(Top,A),FNY(Top,A)
400 PLOT80+4+3,FNX(Bottom,A),FNY(Bottom,A)
410 PROCDROPS(PRECIP)
420 NEXTA
430 MOVEFNX(Top,A2-SPD),FNY(Top,A2-SPD)
440 MOVEFNX(Bottom,A2-SPD),FNY(Bottom,A2-SPD)
450 PLOT80+4+3,FNX(Top,A2),FNY(Top,A2)
460 PLOT80+4+3,FNX(Bottom,A2),FNY(Bottom,A2)
470 ENDPROC
480 :
490 DEFFNX(R,A)=(R-Centre)*SIN(RAD(A))+640
500 :
510 DEFFNY(R,A)=(R-Centre)*COS(RAD(A))+Centre
520 :
```

FOR SALE:

BBC Model B

equipped with:

- Torch pack including twin double sided disc drives
- CPM software to support business software
    - PerfectWriter
    - PerfectCalc
    - PerfectSpell
- Watford 32K Sideways RAM board
- PrintMaster ROM
- Interword ROM
- AMX Mouse

2 Joysticks

High Resolution colour monitor

Some ten discs containing a wide range of games, many never played.

Offers invited:

Contact: Alistair W.
Paraparaumu
Tel (058) 84407

LETTER TO THE EDITOR

I would like to make friendly comment, if I may, on one or two points raised in the Waikato User Group's report (*Beeblet*, April 1988, p.4). I would have enjoyed joining in their discussions.

It was stated in the report that the programs published in *Beeblet* "..are quite long" and the question was asked, "..does anyone type them in?"

Yes, I have used some of the *Beeblet* programs, though certainly not all. I think the longest I have typed in from *Beeblet* was Robert Ngan's "View Printer Driver" (May 1984) of some 250 lines (without the REM's), and the shortest John Andrew's Split Screen Editor (from *Micro User*) "SPLTSCN" (*Beeblet*, Aug 1987). 16 lines, including 6 of instructions. A little beauty! Knowing you, Graham, I gave your one-liner (April 88, p 12) a good hard look, and saved myself the trouble!

I have recently finished typing in a suite of five programs from *Acorn User* (the J.O.E.series) of which one was almost 700 lines. Now to me, that's long! And the debugging was something else! (Took me a month of Sundays to notice the difference between 1 and l -and don't tell me there isn't any!).

So, how long is long? And conversely, how short is short? Does it matter? Not, in my opinion, if the subject matter is of interest, so hopefully a good mix throughout the range from SPLTSCN to *Beebug* FILER....(and J.O.E.?) will arrive at your desk, resulting in a well rounded, full bodied magazine.

As for making the programs available on disc; from a learners point of view I think it would very much reduce the opportunity to learn the finer points of BBC BASIC. And how are we to get our debugging practice? Is the editor to be required to insert a few bugs here and there, with the correct lines entered backwards (upside down?) on the last track? Perhaps a 'short' program could be arranged to spray in a few bugs at random, then the Editor could get some practice too!

Frank Hazlewood,
Auckland.

CONTROL CODES

For use on the Softkeys

By F J Andrew, Wellington

The * commands (including *KEY) are sent direct to the operating system and do not pass through the Tokenising table. As a result, generous use of BASIC routines in the softkeys can quickly use up the little memory normally assigned. BASIC operators and control codes can, however be inserted by combinations of ASCII characters from the tables below.

TABLE 1: CONTROL CODES

KEYS	ASCII CODE		STRING	KEYS	ASCII CODE		STRING
	Dec	Hex			Dec	Hex	
TAB	9	9	¦I	RETURN	13	D	¦M
ESCAPE	27	1B	¦[	DELETE	127	7F	¦?
f0	160	A0	¦! (Space)				
f1	161	A1	¦!!	f2	162	A2	¦!"
f3	163	A3	¦!#	f4	164	A4	¦!$
f5	165	A5	¦!%	f6	166	A6	¦!&
f7	167	A7	¦!'	f8	168	A8	¦!(
f9	169	A9	¦!)				
L-c	172	AC	¦!,	R-c	173	AD	¦!-
D-c	174	AE	¦!.	U-c	175	AF	¦!/
CTL L-c	156	9C	¦! \	CTL R-c	157	9D	¦!]
CTL D-c	158	9E	¦! ^	CTL U-c	159	9F	¦! _
SHF L-c	140	8C	¦! L	SHF R-c	141	8D	¦! M
SHF D-c	142	8E	¦! N	SHF U-c	143	8F	¦! O
CTL A	1	1	¦A	CTL B	2	2	¦B
CTL C	3	3	¦C	CTL D	4	4	¦D
CTL E	5	5	¦E	CTL F	6	6	¦F
.							
CTL etc							
.							
CTL Y	25	19	¦Y	CTL Z	26	1A	¦Z

SHF = Hold the shift key down
CTL = Hold the CTRL key down
L-c = left cursor key
R-c = right cursor key
U-c = up cursor key
D-c = down cursor key

TABLE 2: VDU CODES

(Characters prefixed with ¦)

Keyword	VDU code Dec	 Hex	Char	Keyword	VDU code Dec	 Hex	Char
CLS	12	C	¦L	CLG	16	10	¦P
COLOUR	17	11	¦Q	GCOL	24	18	¦R
MODE	22	16	¦V	PLOT	25	19	¦Y

TABLE 3: BASIC KEYWORDS

(Alphabetic list)

(Requiring character prefix ¦!)

Keyword	Token Hex	Char	Keyword	Token Hex	Char
AUTO	C6	¦!F	BPUT	D5	¦!U
CALL	D6	¦¦V	CHAIN	D7	¦!W
CHR$	BD	¦!=	CLEAR	D8	¦!X
CLG	DA	¦!Z	CLOSE	D9	¦!Y
CLS	DB	¦![	COLOUR	FB	¦!{
DATA	DC	¦!\	DEF	DD	¦!]
DELETE	C7	¦!G	DIM	DE	¦!^
DRAW	DF	¦!_	END	E0	¦!£
ENDPROC	E1	¦!a	ENVELOPE	E2	¦!b
EOF	C5	¦!E			
EVAL	A0	¦! (Space)			
EXP	A1	¦!!	EXT	A2	¦!"
FALSE	A3	¦!#	FN	A4	¦!$
FOR	E3	¦!c	GCOL	E6	¦!f
GET	A5	¦!%	GET$	BE	¦!>
GOSUB	E4	¦!d	GOTO	E5	¦!e
HIMEM	D3	¦!S	IF	E7	¦!g
INKEY	A6	¦!&	INKEY$	BF	¦!?
INPUT	E8	¦!h	INSTR(	A7	¦!'
INT	A8	¦!(	LEFT$(	C0	¦!@
LEN	A9	¦!)	LET	E9	¦!i
LIST	C9	¦!I	LISTO	C9,4F	¦!IO
LN	AA	¦!*	LOAD	C8	¦!H
LOCAL	EA	¦!j	LOG	AB	¦!+
LOMEM	D2	¦!R	MID$(	C1	¦!A
MODE	EB	¦!k	MOVE	EC	¦!I
NEW	CA	¦!J	NEXT	ED	¦!m
NOT	AC	¦!,	OLD	CB	¦!K
ON	EE	¦!n	OPENOUT	AE	¦!.
OPENIN	AD	¦!- (BASIC I)			
OPENUP	AD	¦!- (BASIC II)			

TABLE 3 (Continued): BASIC KEYWORDS
(Alphabetic list)
(Requiring character prefix ¦!)

Keyword	Token Hex	Char	Keyword	Token Hex	Char
PAGE	D0	¦!P	PI	AF	¦!/
PLOT	F0	¦!p	POINT(	B0	¦!0
POS	B1	¦!1	PRINT	F1	¦!q
PROC	F2	¦!r	PTR	CF	¦!O
RAD	B2	¦!2	READ	F3	¦!s
REM	F4	¦!t	RENUMBER	CC	¦!L
REPEAT	F5	¦!u	REPORT	F6	¦!v
RESTORE	F7	¦!w	RETURN	F8	¦!x
RIGHT$(	C2	¦!B	RND	B3	¦!3
RUN	F9	¦!y	SAVE	CD	¦!M
SGN	B4	¦!4	SIN	B5	¦!5
SOUND	D4	¦!T	SQR	B6	¦!6
STOP	FA	¦!z	STR$	C3	¦!C
STRING$(	C4	¦!D	TAN	B7	¦!7
TIME	D1	¦!Q	TO	B8	¦!8
TOP	B8,50	¦!8P	TRUE	B9	¦!9
UNTIL	FD	¦!}	USR	BA	¦!:
VAL	BB	¦!;	VDU	EF	¦!o
VPOS	BC	¦!<	WIDTH	FE	¦!~

TABLE 4 BASIC KEYWORDS
(Alphabetic list)
(Requiring Character prefix ¦!¦)

This list can be inserted using the soft keys .

Key-word	Token (Hex)	Char	Soft Key	Key-word	Token (Hex)	Char	Soft Key
ABS	94	¦!¦T	CTL f4	ACS	95	¦!¦U	CTL f5
ADVAL	96	¦!¦V	CTL f6	AND	80	¦!¦@	SHF f0
ASC	97	¦!¦W	CTL f7	ASN	98	¦!¦X	CTL f8
ATN	99	¦!¦Y	CTL f9	BGET	9A	¦!¦Z	
COS	9B	¦!¦[		COUNT	9C	¦!¦\	
DEG	9D	¦!¦]		DIV	81	¦!¦A	SHF f1
ELSE	8B	¦!¦K		EOR	82	¦!¦B	SHF f2
ERL	9E	¦!¦^		ERR	9F	¦!¦_	
ERROR	85	¦!¦E	SHF f5	HIMEM	93	¦!¦S	CTL f3
LINE	86	¦!¦F	SHF f6	LOMEM	92	¦!¦R	CTL f2
MOD	83	¦!¦C	SHF f3	OFF	87	¦!¦G	SHF f7
OPENIN	8E	¦!¦N	(BASIC II)				

TABLE 4 (Continued): BASIC KEYWORDS
(Alphabetic list)
(Requiring Character prefix ¦!¦)

Key-word	Token (Hex)	Char	Soft Key	Key-word	Token (Hex)	Char	Soft Key
OPT (not available)				OR	84	¦!¦D	SHF f4
PAGE	90	¦!¦P	CTL f0	PTR	8F	¦!¦O	
SPC	89	¦!¦I	SHF f9	STEP	88	¦!¦H	SHF f8
TAB(	8A	¦!¦J		THEN	8C	¦!¦L	
TIME	91	¦!¦q	CTL f1	TRACE	FC	¦!¦¦	

Note : Use of the SHF and CTL with the soft keys will only work on OS 0.1 or later.

SPLIT SCREEN EDITING

by F J Andrew, Wellington

For those that do not have printers, the following program allows the left hand side of a screen to have a display held indefinitely, eg., a catalog. The right hand side can be used to load and list through other programs, etc.

Key in the program, save it and run it.

Press Key f0 and f1 before loading up the display to be held (e.g. *CAT), press Key f2 and then carry out the other activity e.g. load and list of programs.

Pressing key f3 opens up the whole screen.

```
10REM SPLIT SCREEN FOR PROG EDIT
20REM SAVE "SPLTSCN"
30REM M U 11/85 P 175
40*KEY0MODE3¦M
50*KEY1¦\¦@¦'¦@¦L
60*KEY2¦\(¦XO¦@¦L
70*KEY3¦\¦@¦XO¦@
80END
```

LOCAL USER GROUP NEWS

WELLINGTON USER GROUP

The Wellington User Group met for its monthly meeting on the 20th of April to hear a talk about MODEMs. It mostly centred around STARNET using Pace Commstar on a BBC Master and was ably demonstrated by John Andrew, although a few problems were had with the configuration of the Master.

Neil Gordon then demonstrated a 1200 baud full duplex modem (about the size of a pack of cards) which he used to access the Meteorological Office and obtained some impressive weathermaps of New Zealand.

FORWARD PROGRAMME:

The program for the next couple of months is as follows:

15 June:	The ARCHIMEDES will be demonstrated by Neville Hudson with focus going on the new software that is coming out all the time for this marvellous machine.
13 July:	Demonstration of a number of word-processors, including INTERWORD, VIEW and others. This will be followed in corners in which each word-processor will be demonstrated fully.
10 August:	Peter Revell from Barsons will present Education software on the Beeb.

WORKSHOPS:

Please note the change of venue, date and time!!! They are now held on the fourth Tuesday of the month in Penthouse 601 in Wellington College at 7.15pm. To get there go into the foyer of the main building, take the lift to the fifth floor, walk down the corridor, up the stairs. and into Penthouse 601. There are only six computers so it is suggested that you bring your own, or at least your own drives.

DATABASE SERVICES

COMMSTAR 2	Videotex/ASCII communications software for all models of BBC	$ 99.00
TELEPHONE ACCESSORIES	Jackpoints, dual adapters, extension leads, US to BT conversion cables from	$ 9.50
MODEMS	MD312 Manual dial, multispeed 300/300 1200/75 - Telecom approved	$ 375.00
	DISCOVERY 1200C Autodial multispeed 300/300, 1200/1200 - Telecom approved	$ 759.00

Contact:

DATABASE SERVICES

PO BOX 94, AUCKLAND.
STARNET: 9:DRV001

11 ANZAC AVE., AUCKLAND
Telephone: (09) 799658

~~~~~~

For Sale:

NOVEX Hi resolution anmber monitor $150 ONO

Contact: Neil Gordon .. phone (04) 33 8450

****

MEMBERSHIP

Membership of the National User Group is on payment of an annual subscription from April to March, which is $20.00 for the 1988/9 year. A local branch levy may be payable which entitles attendance at meetings as detailed on the back page. Joining or renewing members get all the back issues of *Beeblet* for the current year - those joining after 1 October may have half the annual subscription waived and only receive *Beeblet* from 1 October.

****
~~~~~~

USER GROUP MEETINGS

AUCKLAND: -2nd and 3rd Wednesday of each month at Social Club, Auckland Hospital, Grafton Road at 7.30pm. Contact John Buchanan (Secy), phone 546-985; David Wilkin (Pres), phone 437-540.

North Shore: -last Wednesday of the month, 7.30pm, Salvation Army Hall, Glenfield Road, Glenfield. Contact: Chris, Phone 419-0543 (wk).

CHRISTCHURCH: -fortnightly workshops, Monday 6.30pm at Hagley High School. phone Michael 582-267.

DUNEDIN: -first Thursday of the month. Contact: Martin McDowall, Phone 42-031

HAWKES BAY: -2nd Sunday 2.30pm, Taradale Primary School. Contact Ron: 444-970, John: 445-729 (librarian), Major: 447-729, Mitch: 85-395.

INVERCARGILL: -Southland Education Centre, 70 Doon St. Contact Bob Evans, 73-050 or Lyndon McEntee (020) 87-365.

NELSON: -second Monday of the month, 7.30pm, Nelson Baptist Church, Bridge St. Contact: Peter, Richmond 8975; Ken, Nelson 85-150; Richard, Richmond 8489.

PALMERSTON NORTH:-Phone Brian Knowles 79-319.

ROTORUA: -first Tuesday of month, Science Lab at Rotorua Girls High School. Phone: Ian 85-583, Joan 81-536 Tony 89-316.

TAUPO: -every second Wednesday, 7.00pm. Contact: David 84-215 (h), 85-124 (wk).

TAURANGA: -every second Wednesday. ph. Chris & Jane 65-076.

TIMARU: -Phone Lloyd van der Krogt Timaru 61-412.

TOKOROA: -1st Friday of month, 7.30pm, Tokoroa High School. Contacts: Don Fraser, 68-813 or Graeme Robertson, 65-346 Tokoroa.

WAIKATO: -first Monday of the month, 7.30pm, Waikato Tech Institute. Ph. Alison or John, Morrinsville 6695.

WELLINGTON: -usually second Wednesday, 7.30pm, Royal Society Lecture Theatre, Turnbull St, Thorndon.
-Workshops: fourth Tuesday of month, other details: ph. Don 848-235 or Mike 785-437.

OTHER CENTRES: -let us know and we will publish details here.

BEEBLET

PUBLISHED: monthly except January and mailed to financial members.

CONTRIBUTIONS: -most welcome. Listings should be sent in on tape or disc. (Tape - 2 copies, one at 300 baud please). Please include written explanation of listing; text and graphics only material welcome too.

DEADLINES: -For a particular issue, the last day for material is the fifteenth day of the month prior to publication.

ADVERTISING: -No fixed charge is made for advertising. Supply camera ready, deadlines as above.

Copyright (c) BBC/ACORN COMPUTER USER GROUP OF NZ INC.

Registered Publication with NZPost

THE BBC/ACORN COMPUTER USER GROUP OF NZ INC, PO Box 9592, WELLINGTON

Reproduced by SANDIFORD PRINTERS